Subject: Re: For loops vs. matrix operations
Posted by Craig Markwardt on Wed, 17 Dec 2003 22:57:52 GMT
View Forum Message <> Reply to Message

"Jonathan Greenberg" <greenberg@ucdavis.edu> writes:

> I know some matrix programs perform better if you do straigh matrix math vs.
> a for-next loop -- is idl this way? E.g. is:
>
> array=intarr(10000)
> for i=0,(10000-1) do begin
> array[i]=array[i]+1
> endfor
>
> MUCH slower than:
>
> array=intarr(10000)
> array=array+1
>
> ?
>
> I'm trying to figure out how much time I should be using rewriting some code
> to optimize the algorithm, which is why I'm asking (the code is more complex
> than above, obviously, but I did notice I could "matricize" some of the code
> in places)...

The simplest answer is... optimize the slowest parts. To be a little more specific, the slowest parts are usually the innermost loops, which in your case above *is* the loop. If you can find obvious things like the one you listed above, then definitely do it.

One nice feature of IDL which I didn't know about until recently is PROFILER. While it doesn't give a line-by-line breakdown of execution time, it does give a function-by-function one. If you have more than a few routines, PROFILER should be able to tell you where to start optimizing first.

Happy optimizing!
Craig

--
---------- ----------
Craig B. Markwardt, Ph.D. EMAIL: craigmnet@REMOVEcow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
---------- ----------